REMARKS

This Amendment is in response to the Office Action mailed February 11, 2002. Claims 1-18 and 24-40 are pending, of which Claims 1, 19, 24, and 36-40 are independent. The Applicants notes with appreciation that Claims 19-23 have been allowed. The remaining Claims 1-18 and 24-40 were rejected under 35 U.S.C. § 112, second paragraph and/or under 35 U.S.C. § 103. In response, certain claims have been amended to claim the invention more distinctly and to narrow the issues for appeal. These amendments are not in acquiescence to the rejections. For the reasons stated below, it is believed that the claims are now in condition for allowance.

Rejection under 35 U.S.C. § 112, Second Paragraph

Claims 1, 24, and 36-40 were rejected under 35 U.S.C. § 112, second paragraph. According to the Examiner, the limitation "such that associated values are likely to be retrieved simultaneously" is not definitive. Also, the Examiner notes that he is unable to locate a description for this limitation.

To address this rejection with respect to Claims 1, 24, 36, 38, and 40, the Applicants have amended these claims to include the precise and definite language used in the originally-filed application to describe this feature of the invention. For example, Claims 1, 24, 36, 38, and 40 are amended to replace "likely to be retrieved simultaneously" with "retrievable by a single fetch operation." Support for these amendments can be found in the application, as originally filed, for example, *see* pg. 3, ll. 20-27; pg. 4, ll. 2-4; pg. 11, ll. 11-29; and the Abstract. Thus, no new matter is being introduced. Acceptance is respectfully requested.

In addition, Claims 36 and 39 are amended to remove the limitation "likely to be retrieved simultaneously" from the claim entirely. To expedite the prosecution of these claims, they are further amended to include the limitations of allowed Claim 19. Acceptance is respectfully requested.

It is believed that the amendments to Claims 1, 24 and 36-40 address the rejection under 35 U.S.C. § 112, second paragraph. Entry of the amendments would therefore obviate the rejection and reduce the issues for appeal. Accordingly, it is respectfully requested that this rejection be withdrawn.

Rejection under 35 U.S.C. § 103(a)

Claims 1-18 and 24-40 were rejected under 35 U.S.C. § 103 based on U.S. Patent No. 5,943,677 to Hicks and U.S. Patent No. 6,493,718 to Petculescu. This rejection is traversed.

The method of Claim 1, as amended, is directed to storing data values in a multidimensional database. A plurality of dimensions indicative of storage locations are identified, and a hierarchy of attributes within at least one of the dimensions is identified. Each hierarchy is indicative of an association between the attributes. The attributes are attributed values, and the values are stored based on the values of the hierarchy. In particular, the values are stored such that associated values are retrievable by a single fetch operation. Claims 24, 36, 38 and 40 recite similar limitations.

By storing associated values in such a way that allows them to be retrieved by a single fetch operation, the invention can avoid a situation where there are multiple retrievals for associated values. Associated values, for example, can be stored together on the same disk page. In this way, when performing an aggregation operation, all associated values can be retrieved in a single fetch. In the prior systems, however, "[m]ultiple and often redundant fetches are performed to fetch values stored on the same disk page. These additional fetches increase the time and resources required to complete the aggregation operation. In a large multidimensional database, such increases can be substantial." *See* Specification, pg. 3, ll. 9-12.

The cited references are similar to these prior systems. Hicks, for example, is directed to sparsity management, and discusses an approach for managing sparsity by combining sparse dimensions into composite dimensions. Although Hicks mentions that values that are frequently accessed together are stored near each other, Hicks does not suggest that associated values should be retrieved by a single fetch operation. In particular, the concept of storing values frequently accessed near each other, as described in Hicks, effectively minimizes the distance the disk head has to move to access these values. This concept by its self, however, does not reduce the number of retrievals or fetches. In this way, Hicks does not address the problems associated with multiple and redundant fetches when retrieving associated values stored on the same disk page, or suggest the solutions presented in Claims 1, 24, 36, 38 and 40.

Furthermore, as noted by the Examiner, Hicks does not appear to discuss the claimed hierarchy, which indicates an association between attributes. The Examiner cited Petculescu to show this feature. Petculescu relates to an adaptive caching scheme. Petculescu uses a cost benefit analysis to predict what data cells in a hierarchy of a dimension will be needed to satisfy future queries. In Petculescu, a data retrieval module is used that can balance the benefit of satisfying future queries with the cost of retrieving large data sets from the database. Thus, Petculescu is directed to an adaptive algorithm for retrieving data that can retrieve hierarchies in a dimension. Petculescu, however, does not discuss a technique for storing the data once retrieved. Petculescu only explains that data is stored in the cache for satisfying future queries. As such, Petculescu is not directed to the *storing the data values on a storage medium based on the data values indicated by the hierarchy, such that associated data values are retrievable by a single fetch operation*, as required by Claim 1.

Thus, the resulting combination of Hicks and Petculescu does not show or suggest the limitations of Claim 1. The references taken alone or in combination do not relate the claimed storing data values such that they are retrievable by a single fetch operation. Because the cited references do not discuss the limitations set forth in Claims 1, 24, 36, 38 and 40, it is respectfully requested that the rejection of Claims 1, 24, 36, 38, and 40 under 35 U.S.C. § 103 be withdrawn.

Claims 2-18, which depend from Claim 1, and Claims 25-35, which depend from Claim 24, were rejected under 35 U.S.C. § 103 based on Hicks and Petculescu. These rejections are traversed. Because the requirements of Claims 1 and 24 have been shown above to be absent from the teachings of Hicks and Petculescu, taken alone or in combination, then the limitations of each of the dependent claims are also not in the references. Accordingly, it is respectfully requested that the rejection to dependent Claims 2-18 and 25-35 be withdrawn.

Addressing the rejection to Claims 37 and 39, as discussed above, these claims are amended include the limitations of allowed Claim 19. Accordingly, it is respectfully requested that the rejection to Claims 37 and 39 be withdrawn.

CONCLUSION

In view of the above amendments and remarks, it is believed that all claims are in condition for allowance, and it is respectfully requested that the application be passed to issue. If the Examiner feels that a telephone conference would expedite prosecution of this case, the Examiner is invited to call the undersigned attorney at (978) 341-0036.

Respectfully submitted,

HAMILTON, BROOK, SMITH & REYNOLDS, P.C.

By 

Rodney D. Johnson
Registration No. 36,558
Telephone: (978) 341-0036
Facsimile: (978) 341-0136

Concord, MA 01742-9133
Dated: July 11, 2003

MARKED UP VERSION OF AMENDMENTS

Claim Amendments Under 37 C.F.R. § 1.121(c)(1)(ii)

1. (Twice Amended) A method of storing data values in a multidimensional database comprising:

identifying a plurality of dimensions, wherein each of the dimensions is indicative of a plurality of storage locations;

identifying a hierarchy of attributes within at least one of the dimensions, wherein the hierarchy is indicative of an association between the attributes;

attributing a plurality of data values to each of the attributes; and

storing the data values on a storage medium based on the data values indicated by the hierarchy, such that associated values are retrievable by a single fetch operation. [likely to be retrieved simultaneously.]

24. (Twice Amended) A system for storing and accessing a multidimensional database comprising:

a memory having a cache and a database engine;

a mass storage device in communication with the memory and operable to store a plurality of data values;

a kernel included in the database engine, wherein the kernel is operable to manipulate data values between the memory, the cache, and the mass storage device; and

a sparsity manager included in the database engine, wherein the sparsity manager is operable to determine a storage organization of the data values from a predetermined hierarchy such that associated values are retrievable by a single fetch operation. [likely to be retrieved simultaneously.]

36. (Twice Amended) A computer program product having computer program code for storing data values in a multidimensional database comprising:

computer program code for identifying a plurality of dimensions, wherein each of the dimensions is indicative of a plurality of storage locations;

computer program code for identifying a hierarchy of attributes within at least one of the dimensions, wherein the hierarchy is indicative of an association between the attributes;

computer program code for attributing a plurality of data values to each of the attributes; and

computer program code for storing the data values on a storage medium in proximity to associated data values, wherein the associated data values are attributed to associated attributes as indicated by the hierarchy, such that associated values are retrievable by a single fetch operation. [likely to be retrieved simultaneously.]

37. (Twice Amended) A computer data signal having program code for storing data values in a multidimensional database comprising:

program code for identifying a plurality of dimensions, wherein each of the dimensions is indicative of a plurality of storage locations;

program code for identifying a hierarchy of attributes within at least one of the dimensions, wherein the hierarchy is indicative of an association between the attributes;

program code for attributing a plurality of data values to each of the attributes; [and]

program code for storing the data values on a storage medium in proximity to associated data values, wherein the associated data values are attributed to associated attributes as indicated by the hierarchy, [such that associated values are likely to be retrieved simultaneously.] the data values further including aggregate values and detail values;

program code for aggregating at least one of the dimensions having a hierarchy by traversing each of the aggregate values included in the dimension; and

program code for including, in an aggregation total, the associated data values corresponding to the aggregate value.

38. (Twice Amended) A system for storing data values in a multidimensional database comprising:

means for identifying a plurality of dimensions, wherein each of the dimensions is indicative of a plurality of storage locations;

means for identifying a hierarchy of attributes within at least one of the dimensions, wherein the hierarchy is indicative of an association between the attributes;

means for attributing a plurality of data values to each of the attributes; and

means for storing the data values on a storage medium in proximity to associated data values, wherein the associated data values are attributed to associated attributes as indicated by the hierarchy, such that associated values are retrievable by a single fetch operation. [likely to be retrieved simultaneously.]

39. (Twice Amended) A method of storing data values in a multidimensional database comprising:

identifying a plurality of dimensions, wherein each of the dimensions is indicative of a plurality of attributes associated with a data value;

identifying a hierarchy within at least one of the dimensions, wherein the hierarchy is indicative of an association between the plurality of attributes;

assigning a plurality of data values to each of the plurality of attributes;

storing the data values on a storage medium in proximity to associated data values, wherein the associated data values are assigned to associated attributes as indicated by the hierarchy, [such that associated values are likely to be retrieved simultaneously], the data values further including aggregate values and detail values;

aggregating at least one of the dimensions having a hierarchy by traversing each of the aggregate values included in the dimension; and

including, in an aggregation total, the associated data values corresponding to the aggregate value.

40. (Twice Amended) A method of storing data values in a multidimensional database comprising:

identifying a plurality of dimensions, wherein each of the dimensions is indicative of a plurality of storage locations;

identifying a hierarchy of attributes within at least one of the dimensions, wherein the hierarchy is indicative of an association between the attributes;

attributing a plurality of data values to each of the attributes; and

storing the data values on a storage medium on the same disk page as associated data values such that associated values are retrievable by a single fetch operation [likely to be retrieved simultaneously], wherein the associated data values are attributed to associated attributes as indicated by the hierarchy.